Preprint ANL/MCS-P1796-1010

Extrapolated Multirate Methods for Differential Equations with Multiple Time Scales[*]

Emil M. Constantinescu[†]and Adrian Sandu[‡]

Abstract

In this paper we construct extrapolated multirate discretization methods that allows one to efficiently solve problems that have components with different dynamics. This approach is suited for the time integration of multiscale ordinary and partial differential equations and provides highly accurate discretizations. We analyze the linear stability properties of the multirate explicit and linearly implicit extrapolated methods. Numerical results with multiscale ODEs illustrate the theoretical findings.

Keywords multirate time integration, extrapolation methods, multiscale, linear stability

1 Introduction

In this study we develop multirate time integration schemes using extrapolation methods for efficient simulation of multiscale ODE and PDE problems. In multirate time integration, the time step can vary across the solution components (e.g., spatial domain) and has to satisfy only the local stability conditions, resulting in substantially more efficient overall computations. For PDEs the methods discussed in this paper can be used in the method of lines (MOL) framework, where the temporal and spatial discretizations are independent.

The development of multirate integration is challenging because of the consistency and stability constraints that must be satisfied by the time-stepping

[*] Emil Constantinescu was supported in part by the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract DE-AC02-06CH11357, and by the National Science Foundation through award NSF CCF-0515170. The work of Adrian Sandu was supported in part by NSF through the awards NSF CCF-0515170, NSF OCI-0904397, NSF CCF-0916493, and NSF DMS-0915047.

[†] Mathematics and Computer Science Division, Argonne National Laboratory, 9700 S Cass Avenue, Argonne, IL 60439, USA; emconsta@mcs.anl.gov

[‡] Department of Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061, USA; asandu@cs.vt.edu

schemes. Engstler and Lubich [12] developed multirate schemes based on extrapolated forward Euler methods (MURX). The components with slow dynamics are inactivated at certain time levels, while the fast components are evaluated every time step. Our work extends this strategy to extrapolated explicit and implicit compound multirate steps. In this case the extrapolation procedure operates on multirate time stepping schemes.

The concept of multirate methods was introduced in such studies as [27; 14; 17; 33], and more recent results are presented in [16; 23; 22; 2; 4; 28; 6]. Multirate methods for conservative laws are developed in [4; 28; 8; 21; 34; 32] and for parabolic equations using a locally self-adjusting multirate time-stepping scheme in [30; 31].

In this paper we are concerned with solving the following initial value problem,

$$\begin{aligned}
&\mathbf{y}'(x) = f(x, \mathbf{y}(x))\,,\ x > x_0\,,\ \mathbf{y}(x_0) = y_0\,, y \in \mathbb{R}^D \text{ with} \qquad (1)\\
&\mathbf{y} = [y_1\ y_2\ \ldots\ y_M]^T\ ,\ f(x,\mathbf{y}) = [f_1(x,\mathbf{y})\ f_2(x,\mathbf{y})\ \ldots\ f_M(x,\mathbf{y})]^T \text{ and}\\
&y_r \in \mathbb{R}^{d_r}\,,\ f_r : \mathbb{R}^{D+1} \to \mathbb{R}^{d_r}\,,\ r = 1, \ldots, M\,,\ \text{and} \sum_{r=1}^{M} d_r = D\,,
\end{aligned}$$

where $\mathbf{y}$ is the solution vector partitioned into components y_r, $r = 1, \ldots, M$, that have their own time scales. Among others, these types of problems occur naturally in electric circuit simulations [2] and in problems using variable grid sizes [7]. We seek to apply time discretization methods with a different time-step length for each dynamic characteristic to (1). To this end we consider extrapolation methods [10; 18; 19] with multirate explicit and implicit base schemes for time marching. When solving space-/time-dependent PDEs in the method of lines framework, we use f to represent the spatial discretization operator.

For simplicity, but without loss of generality, we consider the simplified two-scale problem

$$\begin{cases} y'(x) = f\left(x, y(x), z(x)\right) \\ z'(x) = g\left(x, y(x), z(x)\right) \end{cases} \quad [y(x_0)\ z(x_0)]^T = [y_0\ z_0]^T\ ,\ x > x_0\,, \qquad (2)$$

where y represents the slowly evolving components and z the fast ones. Form (1) is obtained immediately by successive refinements of (2). We assume that such partitions are relatively easy to obtain algorithmically. We illustrate such a multirate strategy on a practical problem in Section 5.

Multirate methods are typically more efficient than the "classical" single-rate methods by avoiding the global time-step restrictions imposed by stability or accuracy considerations. The efficiency gains can be accurately estimated if the computational work in evaluating the individual components of $f(x, \mathbf{y})$ is proportional to n_r, $r = 1, \ldots, M$, which is typically the case in practice. For instance, if we consider problem (2) with equally sized y and z components and with the fast dynamics being five times faster than the slow dynamics, the computational cost is about 40% less with a multirate scheme than with its corresponding single-rate implementations.

The rest of this paper is organized as follows. In the next section we review the extrapolation methods. In Sec. 3 we introduce several multirate base methods, the main result of this paper. Linear stability of the proposed schemes is analyzed in Sec. 4, and numerical examples are given in Sec. 5. We present our conclusions in Sec. 6.

2 Extrapolation Methods

Extrapolation methods were introduced in their present form by Gragg and Stetter [15] as a simple way to obtain high-order discretizations of initial value problems. Here we briefly describe the basic ideas behind these methods. Consider a sequence n_i of positive integers with $n_i < n_{i+1}$, $1 \leq i \leq E$, and define the corresponding step sizes $h_1, h_2, h_3, \ldots$ by $h_i = H/n_i$. Further, define the numerical approximation of (1) at $x_0 + H$, using the step size h_i, by

$$T_{i,1} := y_{h_i}(x_0 + H)\,,\ 1 \leq i \leq E\,. \tag{3}$$

This approximation is obtained by using a *base method*. Let us assume that the local error of the pth-order method used to solve (3) has an asymptotic expansion of the form

$$y(x) - y_h(x) = e_{p+1}(x)\,h^{p+1} + \cdots + e_N(x)\,h^N + Err_h(x)\,h^{N+1}\,, \tag{4}$$

where $y(x)$ is the exact solution at x, $e_i(x)$ are errors that do not depend on h, and Err_h is bounded for $x_0 \leq x \leq x_{\text{end}}$. By using E approximations to (3) with different h_i's, one can eliminate the error terms in the local error asymptotic expansion (4) by employing the same procedure as in Richardson extrapolation (see [18, Chap. II.9]). High-order approximations of the numerical solution of (1)-(2) can be determined by solving a linear system with E equations. Then the kth solution represents a numerical method of order $p + k - 1$ [18, Chap. II, Thm. 9.1]. The most economical solution to this set of linear equations is given by the Aitken-Neville formula [1; 24; 13]:

$$T_{j,k+1} = T_{j,k} + \frac{T_{j,k} - T_{j-1,k}}{(n_j/n_{j-k}) - 1}\,,\ j = 2, 3, \ldots\,,\quad k = 1, \ldots, j-1. \tag{5a}$$

If the numerical method (3) is symmetric (see [18, Def II.8.7]; e.g., implicit trapezoidal or mid-point rules), then the Aitken-Neville formula yields

$$T_{j,k+1} = T_{j,k} + \frac{T_{j,k} - T_{j-1,k}}{(n_j/n_{j-k})^2 - 1}\,,\ j = 2, 3, \ldots\,,\quad k = 1, \ldots, j-1. \tag{5b}$$

Scheme (3), (5) is called the *extrapolation method*. For illustration purposes, the $T_{j,k}$ solutions can be represented in a tableau (1). As can be seen from

Table 1: Tableaux with (a) the $T_{j,k}$ solutions and (b) their corresponding orders using (5a).

T_{11}			
T_{21}	T_{22}		
T_{31}	T_{32}	T_{33}	
$\cdots$	$\cdots$	$\cdots$	$\cdots$

(a) Extrapolation $T_{j,k}$ tableau

p			
p	$p+1$		
p	$p+1$	$p+2$	
$\cdots$	$\cdots$	$\cdots$	$\cdots$

(b) Orders of $T_{j,k}$ using (5a)

the tableau, the method is represented by a sequence of consistent embedded methods that can be used for step-size control and variable-order approaches. One has several choices for the sequences n_j, Deuflhard [9] however, showed that the harmonic sequence $n_j = 1, 2, 3, \ldots$ is the most economical, and this sequence therefore will be used for the rest of this study.

The base methods are typically chosen to be low-order schemes with $p = 1, 2$, such as explicit Euler, linearly implicit Euler, or trapezoidal rule, because of efficiency considerations [18]. Linearly implicit Euler is a linearization of the implicit Euler scheme. Under typical smoothness assumptions, one has

$$\mathbf{y}_{i+1} = \mathbf{y}_i + hf\left(\mathbf{y}_{i+1}\right) \approx \mathbf{y}_i + h\left(f\left(\mathbf{y}_i\right) + J\left(\mathbf{y}_{i+1} - \mathbf{y}_i\right)\right) ,$$

where $\mathbf{y}_i$ is the numerical solution at time index i, and J is a consistent approximation to $\frac{\partial f(\mathbf{y}_i)}{\partial \mathbf{y}}$. For brevity, the time dependence of the right-hand side is not illustrated. Then the *linearly implicit Euler* method is given by

$$\left(I - hJ\right)\left(\mathbf{y}_{i+1} - \mathbf{y}_i\right) = hf\left(\mathbf{y}_i\right) . \tag{6}$$

This method has been used in [10; 11] as the base method for solving stiff ODEs of type (1) with the extrapolation method (3), (5).

Extrapolation methods can be easily parallelized. Notice that the terms in the first column of Tableau 1a, which represent the bulk of the computation, are independent of each other [26; 3; 5] and have predictable costs. Utilizing these facts yields robust implementations on current computational architectures by optimally scheduling each tableau row to be solved on a given set of processing units. Moreover, because of their simple construction, the extrapolation methods can easily provide solutions with arbitrary orders of accuracy. Higher orders are obtained by computing more entries in the tableau.

3 Multirate Base Methods

We consider three multirate base methods for solving (2) with the extrapolation algorithm (3), (5). Our approach extends trivially to general partitions such as

(1). We begin with the $m-$rate *multirate explicit Euler method*,

$$\left\{\begin{aligned} & y_{n+1} = y_n + h\, f(y_n, z_n) \\ & z_{n+\frac{i}{m}} = z_{n+\frac{i-1}{m}} + \frac{h}{m}\, g(Y_{n+\frac{i-1}{m}}, z_{n+\frac{i-1}{m}})\,,\ i = 1, \ldots, m\,, \end{aligned}\right. \tag{7}$$

where m is a positive integer and $Y_{n+\frac{i}{m}}$ is an approximation of y at $x_{n+\frac{i}{m}}$. Forward Euler is first-order accurate, and hence zeroth-order interpolation is suited to approximate Y: $Y_{n+\frac{i}{m}} = y_n$ or $Y_{n+\frac{i}{m}} = y_{n+1}$; by using the former, a more parallelizable implementation may be obtained. The first-order interpolation can also be considered: $Y_{n+\frac{i-1}{m}} = \frac{m-i+1}{m} y_n + \frac{i-1}{m} y_{n+1}$, $i = 1, \ldots, m$. Formally we have

$$Y_{n+\frac{i-1}{m}} = y_n\,, \tag{8a}$$

$$Y_{n+\frac{i-1}{m}} = y_{n+1}\,, \tag{8b}$$

$$Y_{n+\frac{i-1}{m}} = \frac{m-i+1}{m} y_n + \frac{i-1}{m} y_{n+1}\,. \tag{8c}$$

All three possibilities are considered in this study.

The linearly implicit Euler method (6) can also be considered as a candidate for the base methods used in the extrapolation procedure. In this case, two multirate methods are proposed: *slowest-first* and *compound*. The slowest-first m-rate *multirate linearly implicit method* is given by

$$\left\{\begin{aligned} & \begin{bmatrix} I - hf_y(0) & -hf_z(0) \\ -hg_y(0) & I - hg_z(0) \end{bmatrix} \cdot \begin{bmatrix} y_{n+1} - y_n \\ z_{n+1} - z_n \end{bmatrix} = \begin{bmatrix} h\, f\,(y_n, z_n) \\ h\, g\,(y_n, z_n) \end{bmatrix}, \\ & \left(I - \tfrac{h}{m} g_z(0)\right) \left(z_{n+\frac{i}{m}} - z_{n+\frac{i-1}{m}}\right) = \tfrac{h}{m}\, g\left(Y_{n+\frac{i-1}{m}}, z_{n+\frac{i-1}{m}}\right), \\ & \qquad\qquad\qquad\qquad i = 1, \ldots, m\,, \end{aligned}\right. \tag{9}$$

where the shorthand notation $f_{\{y\,,z\}}(0)$ and $g_{\{y\,,z\}}(0)$ denotes the derivatives evaluated at x_0, that is, the beginning of the current extrapolation time step in (3). Another strategy is to advance the solution components with their respective rates leading to the compound m-rate *multirate linearly implicit method*:

$$\left\{\begin{aligned} & \begin{bmatrix} I - hf_y(0) & -hf_z(0) \\ -\dfrac{h}{m} g_y(0) & I - \dfrac{h}{m} g_z(0) \end{bmatrix} \cdot \begin{bmatrix} y_{n+1} - y_n \\ z_{n+\frac{1}{m}} - z_n \end{bmatrix} = \begin{bmatrix} h\, f\,(y_n, z_n) \\ \dfrac{h}{m}\, g\,(y_n, z_n) \end{bmatrix}, \\ & \left(I - \tfrac{h}{m} g_z(0)\right) \left(z_{n+\frac{i}{m}} - z_{n+\frac{i-1}{m}}\right) = \tfrac{h}{m}\, g\left(Y_{n+\frac{i-1}{m}}, z_{n+\frac{i-1}{m}}\right), \\ & \qquad\qquad\qquad\qquad i = 2, \ldots, m\,. \end{aligned}\right. \tag{10}$$

Just as for the explicit case, in the last two schemes $Y_{n+\frac{i}{m}}$ is obtained from (8).

We propose to use the first-order accurate multirate schemes (7), (9), and (10) as base methods in the extrapolation procedure. These methods can be shown to possess a local error expansion of form (4) and therefore can be extrapolated by using (3),(5a). It follows that the proposed extrapolated multirate

Table 2: Classical (a) local and (b) global orders for the extrapolation methods with first order base methods.

2			
2	3		
2	3	4	
. . .	. . .	. . .	. . .

(a) Local orders

1			
1	2		
1	2	3	
. . .	. . .	. . .	. . .

(b) Global orders

methods' orders of convergence are accurately captured in Table 2. This fact is also verified experimentally in Sec. 4. In terms of the computational cost between (9) and (10), method (10) has a slight advantage because the fast variables are evolved in one step less. We note here the differences between the methods proposed in this study and Engstler and Lubich's explicit MURX [12] schemes. In our approach the variable partition is performed ab initio, and each time step calculation is multirate in itself, whereas in MURX the slow computations cease upon reaching an error criterion. In other words, MURX has a slow and a fast tableau, whereas the proposed multirate schemes partition the very base method into slow and fast parts. The extrapolated methods based on (7), (9), and (10) can easily scheduled on parallel machines, as illustrated in [26; 5] because of their uniform structure.

Following [12] we note that by construction the extrapolation methods provide automatic lower-order embedded methods, a fact also revealed by a quick inspection of Tableaux 2. This can provide automatic error control mechanisms such as the ones implemented in [12]. This aspect will not be further addressed in this study; we refer the reader to [12] for more details instead.

Next we illustrate the theoretical linear stability and accuracy results on numerical examples using the extrapolation scheme with base methods (7), (9), and (10).

4 Linear Stability Analysis of the Extrapolated Multirate Methods

Following the analysis done by Kværnø [22], we investigate the extrapolated schemes with the base methods defined by (7) and (9) applied to the following generic linear test problem,

$$\left(\begin{array}{c} \widehat{y}(x) \\ \widehat{z}(x) \end{array} \right)' = \left(\begin{array}{cc} \alpha_{11} & \alpha_{12} \\ \alpha_{21} & \alpha_{22} \end{array} \right) \left(\begin{array}{c} \widehat{y}(x) \\ \widehat{z}(x) \end{array} \right) = \left(\begin{array}{c} f\left(\widehat{y}(x), \widehat{z}(x)\right) \\ g\left(\widehat{y}(x), \widehat{z}(x)\right) \end{array} \right),$$

where $\alpha_{ij} \in \mathbb{R}$. Our implicit assumption is that this system has two underlying time scales. Therefore, to simplify the stability analysis, we consider the scaled system (i.e., $x \to (-\alpha_{11})x$):

$$\left(\begin{array}{c} y(x) \\ z(x) \end{array} \right)' = \underbrace{\left(\begin{array}{cc} -1 & \varepsilon \\ \omega & -m \end{array} \right)}_{A} \left(\begin{array}{c} y(x) \\ z(x) \end{array} \right) = \left(\begin{array}{c} f\left(y(x), z(x)\right) \\ g\left(y(x), z(x)\right) \end{array} \right). \qquad (11)$$

In this scaling we assume for simplicity that m is an integer, and thus we obtain the scale difference (m) between the slow component, y, and the fast one, z. The coupling between these two components is represented by ε and ω. System (11) is stable if the real part of the eigenvalues of A is negative, which gives $\omega\varepsilon \leq m$. In addition, we assume that $|\varepsilon| \leq 1$ and $|\omega| \leq m$, thereby guaranteeing that system (11) has two distinct scales such as (1).

The transfer or stability function depends on the coefficients of A multiplied by the time step and denoted for short by $R(hA)$. For a numerical discretization of (11), the stability function is defined by the quantity that verifies

$$\left(\begin{array}{c} y_{n+1} \\ z_{n+1} \end{array} \right) = R(hA) \left(\begin{array}{c} y_n \\ z_n \end{array} \right).$$

In order for the discretization method to be stable, one needs to have the spectral radius $\rho(R(hA)) \leq 1$. The stability functions of (7) and (9) can be easily calculated. The stability function of the extrapolated method is calculated from the extrapolation formula (5a) as [19, Chap. IV]:

$$R_{j,k+1}(hA) = R_{j,k}(h\dots) + \frac{R_{j,k}(hA) - R_{j-1,k}(hA)}{(n_j/n_{j-k}) - 1}.$$

In our case R is a function of $h\omega$ and $h\varepsilon$, because a consistent linear method is assumed.

We take a practical approach and ask the following question: How does the stability region of a multirate method with ratio m applied to (11) compare with the stability region of the single-rate method with the time-step length of the fastest component (i.e., H/m)? In other words we look for the degradation or appreciation in stability of the multirate method compared with the single-rate method. We note that the multirate method is more efficient in this case because it takes fewer steps on the slow components.

The stability analysis of the proposed methods is tedious and complex, and often we have not been able to obtain closed forms of the stability functions. We therefore investigate numerically the linear stability properties. In particular, we explore the stability of extrapolation method (3), (5a) with the multirate base methods (7), (9), and (10) applied to problem (11). We further consider the ratio $m = 2$ fixed and investigate the stability region ($\rho(R) \leq 1$) in the $h\omega$-$h\varepsilon$ plane. Moreover, to assess the stability properties for large values of either or both arguments, we rescale the axes by $\frac{1}{1+|h\varepsilon|}$ and $\frac{1}{1+|h\omega|}$. Therefore the entire stability region is restricted to domain $(0,1) \times (0,1)$.

4.1 Linear Stability of the Extrapolated Multirate Explicit Euler Method

In Figure 1 we show the stability regions for the extrapolated multirate explicit method (7) for the extrapolation terms in positions T_{22} and T_{54}, (see Table 1). The single-rate stability regions are computed for a time step smaller than the one considered for the multirate results by a factor of the scale ratio. In other words, we expect the multirate methods to be stable if their slow components are integrated with a time step larger by a factor given by the scale than the time step that provides global stability (i.e., stability for the single-rate scheme). The stability region of the multirate method is slightly degraded for T_{22} with (8a); however, the linear interpolation (8c) seems to have a stabilizing effect, and for practical purposes we consider that this reduction in the stability region is acceptable. We note that a fair comparison in terms of efficiency in the multirate case would have the $h\varepsilon$ direction amplified by $\mathfrak{m}$.

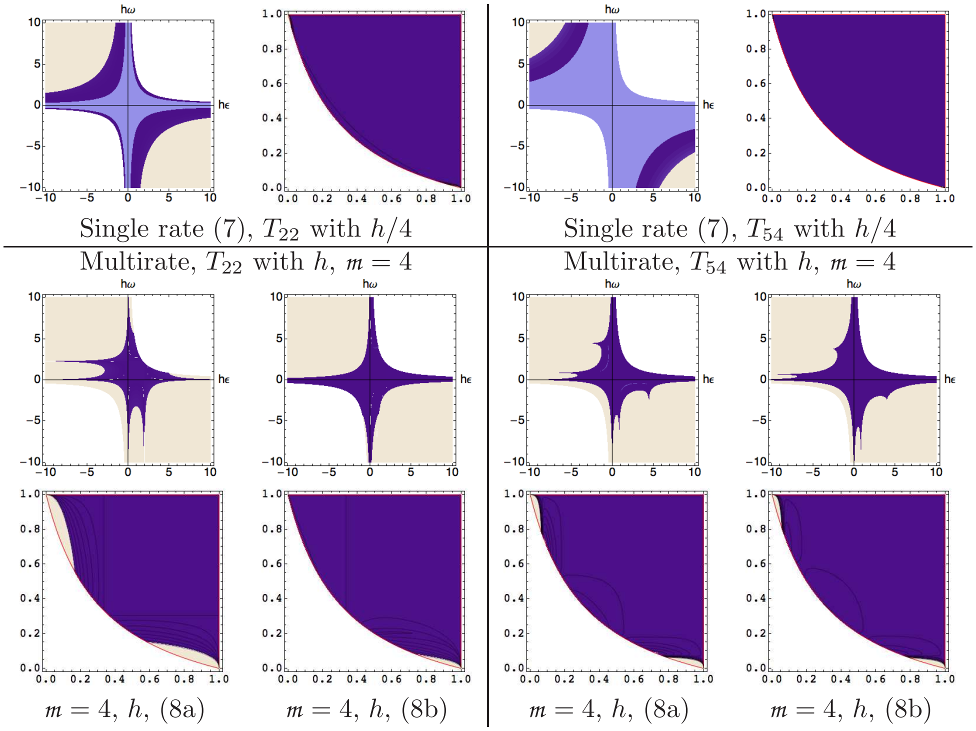
The lack of stability for the multirate scheme in T_{22} can be avoided by considering solutions given by off-diagonal extrapolation terms. Note that, for instance, T_{54} does not use approximation T_{11} that can propagate an unstable solution approximation through the tableau diagonal, and therefore the small step-size computations (i.e., H/k, $k = 2, \ldots, 5$) in T_{54} yield a stable solution.

4.2 Linear Stability of the Extrapolated Multirate Linearly Implicit Methods

We now explore the stability regions of the proposed implicit multirate schemes (9) and (10). We have determined numerically that in the first column of the extrapolation tableau the multirate implicit methods preserve the “unconditional” stability of the implicit base (single-rate) method; in other words, the stability region extends to (∞, ∞) and $(-\infty, -\infty)$ in the $h\omega$-$h\varepsilon$ plane. In Figure 2 we show an analysis similar to the one done in the previous section. We illustrate the stability region of the proposed multirate methods for $|h\omega|$, $|h\varepsilon|$ and also for their scaled versions by $\frac{1}{1+|h\varepsilon|}$ and $\frac{1}{1+|h\omega|}$ that gives one a global, albeit coarse, view of the entire stability region. By using this strategy we conclude that the only multirate strategy that preserves the “unconditional” stability of the single rate method beyond the first column is the slowest first strategy (9) with (8b). In practice, however, we have not noticed any stability issues with the other multirate strategies.

5 Numerical Experiments with Extrapolated Multirate Methods

In this section we illustrate numerical results obtained using the proposed methods. We consider two problems: a multirate equivalent of the classical Prothero-Robinson [25] and what is known as the inverter-chain problem [2].



Single rate (7), T_{22} with $h/4$

Single rate (7), T_{54} with $h/4$

Multirate, T_{22} with h, $m = 4$

Multirate, T_{54} with h, $m = 4$

$m = 4$, h, (8a)

$m = 4$, h, (8b)

$m = 4$, h, (8a)

$m = 4$, h, (8b)

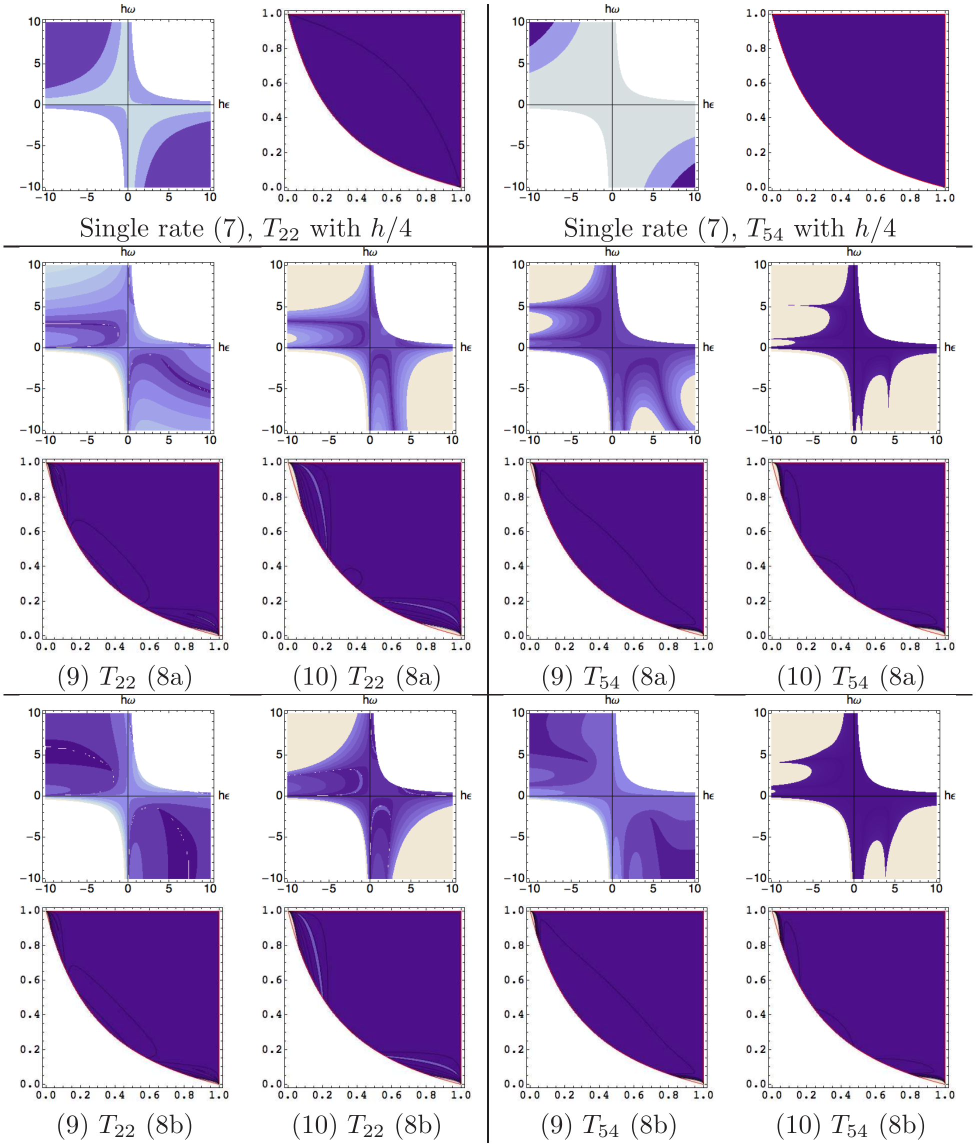
Figure 1: Stability region ($\rho(R)$) for problem (11) with the explicit single rate ($m = 1$) method (7) and the corresponding multirate ($m = 4$) methods for entries T_{22} and T_{54} in the extrapolation tableau. Also shown are the corresponding axes-scaled stability regions by $\frac{1}{1+|h\varepsilon|}$ and $\frac{1}{1+|h\omega|}$.

5.1 Modified Prothero-Robinson Problem

Consider the following linear initial value problem,

$$\begin{aligned}
\begin{pmatrix} \widehat{y}(x) \\ \widehat{z}(x) \end{pmatrix}' &= \begin{pmatrix} \Gamma & \varepsilon \\ \varepsilon & -1 \end{pmatrix} \begin{pmatrix} \widehat{y}(x) - g(x) \\ \widehat{z}(x) - g(\omega x) \end{pmatrix} + \begin{pmatrix} g(x) \\ g(\omega x) \end{pmatrix}', \\
\begin{pmatrix} \widehat{y}(x_0) \\ \widehat{z}(x_0) \end{pmatrix} &= \begin{pmatrix} g(x_0) \\ g(\omega x_0) \end{pmatrix},
\end{aligned}$$

where g is a known function. This problem was adapted to vector form [2] from the scalar Prothero-Robinson [25; 19] test problem. The same offdiagonal value is used to simplify the analysis. The exact solution is $[\widehat{y}(x)\ \widehat{z}(x)]^T = [g(x)\ g(\omega x)]^T$. This form allows us to control the stiffness, coupling, and scale through Γ, ε, and ω, respectively.



Single rate (7), T_{22} with $h/4$

Single rate (7), T_{54} with $h/4$

(9) T_{22} (8a)

(10) T_{22} (8a)

(9) T_{54} (8a)

(10) T_{54} (8a)

(9) T_{22} (8b)

(10) T_{22} (8b)

(9) T_{54} (8b)

(10) T_{54} (8b)

Figure 2: Stability region ($\rho(R)$) for problem (11) with the implicit single-rate ($m = 1$) methods (9) and (10) and the corresponding multirate ($m = 4$) methods for entries T_{22} and T_{54} in the extrapolation tableau. The bottom two rows are methods using (9) and (10) with time step h. Also shown are the corresponding axes-scaled stability regions by $\frac{1}{1+|h\varepsilon|}$ and $\frac{1}{1+|h\omega|}$.

We perform the following change of variables:

$$\begin{pmatrix} \widehat{y}(x) \\ \widehat{z}(x) \end{pmatrix} = \begin{pmatrix} -1 + y^2(x) \\ -2 + z^2(x) \end{pmatrix}, \quad \begin{pmatrix} \widehat{y}(x) \\ \widehat{z}(x) \end{pmatrix}' = \begin{pmatrix} 2y(x)y'(x) \\ 2z(x)z'(x) \end{pmatrix},$$

$$\begin{pmatrix} y(x_0) \\ z(x_0) \end{pmatrix} = \begin{pmatrix} \sqrt{1 + g(x_0)} \\ \sqrt{2 + g(\omega x_0)} \end{pmatrix}.$$

The problem in y and z becomes nonlinear; and if $g(x) = \cos(x)$, the following problem is obtained:

$$\begin{pmatrix} y(x) \\ z(x) \end{pmatrix}' = \begin{pmatrix} \Gamma & \varepsilon \\ \varepsilon & -1 \end{pmatrix} \begin{pmatrix} \left(-1 + y^2 - \cos(x)\right)/(2y) \\ \left(-2 + z^2 - \cos(\omega x)\right)/(2z) \end{pmatrix} - \begin{pmatrix} \sin(x)/(2y) \\ \omega \sin(\omega x)/(2z) \end{pmatrix}. \tag{12a}$$

The exact solution of (12a) is given by given by

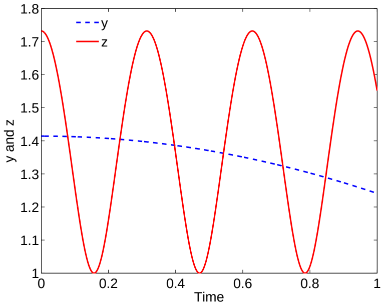
$$\begin{pmatrix} y(x) \\ z(x) \end{pmatrix} = \begin{pmatrix} \sqrt{1 + \cos(x)} \\ \sqrt{2 + \cos(\omega x)} \end{pmatrix} \tag{12b}$$

and represented in Figure 3. We refer to problem (12) as KPR.

The theoretical findings from the preceding sections are illustrated on the KPR problem (12) discretized by using extrapolation procedure (3), (5a) with base methods (7), (9), and (10). The experiments consist in integrating the KPR problem with successively smaller steps H. We begin by exploring the consistency of the proposed methods and set up an experiment using $\varepsilon = 0.5$, $\Gamma = -2.0$, $\omega = 20.0$. The multirate schemes use a rate of $m = \omega = 20$. The observed orders based on the numerical error in L_1 and L_2 norms are presented in Figure 3 and confirm the theoretical expectations as discussed in Section 2.

The accuracy and efficiency of the multirate methods are explored in three parameter settings: nonstiff, stiff, and high frequency. The last setting is used to explore a case in which the scale ratio is higher than the considered rate. The errors are computed by using the exact solution at the final time $T = 0.3$ in the L_2 norm.

The nonstiff case In Table 3 we show the errors for KPR with $\Gamma = -2.0$, $\omega = 5$, and $\varepsilon = 0.05$ using the single rate explicit scheme (7) with $H = 0.01$ and multirate with $m = 5$, $H = 0.05$. The multirate strategy has a smaller computational cost (40% less work) and provides the same level of accuracy. One conclusion from our experiments is that the slow interpolation scheme does not seem to affect the stability of the solution, albeit some theoretical evidence indicates that one should use either (8b) or (8c).



2						
2	3					
2	3	4				
2	3	4	5			
2	3	4	5	6		
2	3	4	5	6	7	
2	3	4	5	6	7	8
...	...	...	...	...	...	...

Figure 3: The exact solution of the modified nonlinear Prothero-Robinson equation (12) with $\varepsilon = 0.5$, $\Gamma = -2.0$, $\omega = 20.0$ (left) and the observed numerical local discretization order of the extrapolation method (3), (5a) with the multirate (two-rate ($m = \omega = 20$)) base method.

Table 3: Errors for the nonstiff ($\Gamma = -2.0e+00$, $\omega = 5.0e+00$, $\varepsilon = 5.0e-02$) KPR problem at the final time ($T = 0.3$) for extrapolation terms up to order five, solved with |SR, $H = 0.01$ |$m = 5$, $H = 0.05$ | using the explicit method (7) with slow interpolation (8a).

7.2e-3	7.6e-3								
3.6e-3	3.8e-3	4.3e-5	4.6e-5						
2.4e-3	2.5e-3	1.4e-5	1.5e-5	2.3e-7	2.9e-7				
1.8e-3	1.9e-3	7.0e-6	7.5e-6	5.7e-8	7.2e-8	8.3e-10	2.1e-9		
1.4e-3	1.5e-3	4.2e-6	4.5e-6	2.3e-8	2.9e-8	1.6e-10	4.1e-10	3.3e-12	2.0e-11

The high-frequency case In the high-frequency scenario we consider $\Gamma = -2.0e+02$, $\omega = 3.0e+01$, $\varepsilon = 5.0e-02$, and use the single-rate explicit scheme (7) with $H = 0.1$ and $H = 0.02$ and multirate with $m = 5$, $H = 0.1$. Several aspects should be noticed in the results presented in Table 4: The single-rate result with $H = 0.1$ is unstable (the error is growing). The single rate with $H = 0.02$ is stable; however, it yields comparable results with the multirate scheme using $H = 0.1$ that requires 40% fewer computations than does the stable single-rate method. This aspect results from the fact that term T_{11} is not stable and therefore all the resulting computations become unstable (see [18; 19; 5]). A simple solution to circumvent this problem is to compute only the off-diagonal extrapolation terms.

The stiff case In Table 5 we show the errors for the KPR problem with $\Gamma = -2.0e+05$, $\omega = 2.0e+01$, $\varepsilon = 5.0e-01$ using the single-rate explicit scheme (9) with $H = 0.025$ and multirate with $m = 4$, $H = 0.1$. With direct linear algebra methods or known Jacobian, the multirate strategy has a smaller computational cost (38% less work) and provides a more accurate solution. Both implicit methods (9) and (10) give similar results, and we therefore present only

Table 4: Errors for the high-frequency ($\Gamma = -2.0e+02$, $\omega = 3.0e+01$, $\varepsilon = 5.0e-02$) KPR problem at the final time ($T = 0.3$) solved with |SR, $H = 0.1$ |SR, $H = 0.02$ |$m = 5$, $H = 0.1$ | using the explicit method (7) with slow interpolation (8a). Term T_{55} =|1.3e+8|1.1e+3|5.3e+0|.

8.5e+1	7.2e+0	7.2e+0									
2.4e+1	1.3e-2	1.3e-2	7.1e+3	8.8e+4	5.2e+1						
1.0e+3	5.6e-3	5.8e-3	1.1e+5	1.3e+2	8.9e-2	5.7e+5	4.2e+3	5.7e+1			
1.0e+2	4.1e-3	4.3e-3	9.3e+4	4.9e-4	4.1e-4	2.7e+6	1.8e+3	7.8e-3	9.7e+6	7.5e+3	1.1e+1
7.2e+0	3.2e-3	3.4e-3	3.1e+4	2.4e-4	2.4e-4	3.7e+6	1.4e-4	1.5e-5	4.5e+7	2.8e+0	3.5e-2

the former. For high-order approximations and very stiff problems one should use only off-diagonal extrapolation terms for computation [19; 5].

Table 5: Errors for the stiff ($\Gamma = -2.0e+05$, $\omega = 2.0e+01$, $\varepsilon = 5.0e-01$) KPR problem at the final time ($T = 0.3$) solved with |SR,$H = 0.025$ |$m = 4$,$H = 0.1$ | using the linearly implicit method (9) with slow interpolation (8a).

8.2e-2	8.5e-2								
3.0e-2	3.1e-2	1.9e-2	1.3e-2						
1.8e-2	1.8e-2	5.0e-3	5.1e-3	1.3e-3	1.2e-3				
1.3e-2	1.3e-2	2.7e-3	2.7e-3	3.3e-4	2.7e-4	9.6e-4	5.5e-5		
9.7e-3	9.9e-3	1.6e-3	1.7e-3	9.6e-5	9.7e-5	5.9e-5	1.9e-5	3.0e-4	9.7e-6

The KPR problem provides a controlled and idealized approach that illustrates the properties and robustnes of multirate methods. We next present experiments with a practical problem.

5.2 Inverter-Chain Problem

Consider the inverter-chain problem [2; 29],

$$\begin{aligned} & y_j'(t) = U_{OP} - y_j(t) - \Upsilon F\left(y_{j-1}(t)), y_j(t))\right)\,,\ j = 1, \ldots, n\,,\ 0 \le t \le T\,, \\ & F(u,v) = \max\left(u - U_{thres}, 0\right)^2 - \max\left(u - v - U_{thres}, 0\right)^2\,, \end{aligned} \tag{13a}$$

$$y_j(0) = \begin{cases} 6.247 \cdot 10^{-3}, & j \text{ even} \\ 5, & j \text{ odd} \end{cases}\,, \quad y_0(t) = \begin{cases} t-5, & 5 \le t \le 10 \\ 5, & 10 \le t \le 15 \\ \frac{5}{2}(17-t), & 15 \le t \le 17 \\ 0, & \text{otherwise} \end{cases}\,, \tag{13b}$$

with the following parameters: $n = 500$ inverters, $U_{OP} = 5$, $U_{thres} = 1$, and $T = 120$ seconds. The Υ coefficient determines whether the problem is relatively stiff ($\Upsilon := 100$) or nonstiff ($\Upsilon := 1$). The solution at $t = 15$ s, $t = 60$ s, $t = 120$ s is illustrated in Figure 4. A detailed analysis of this problem in the multirate

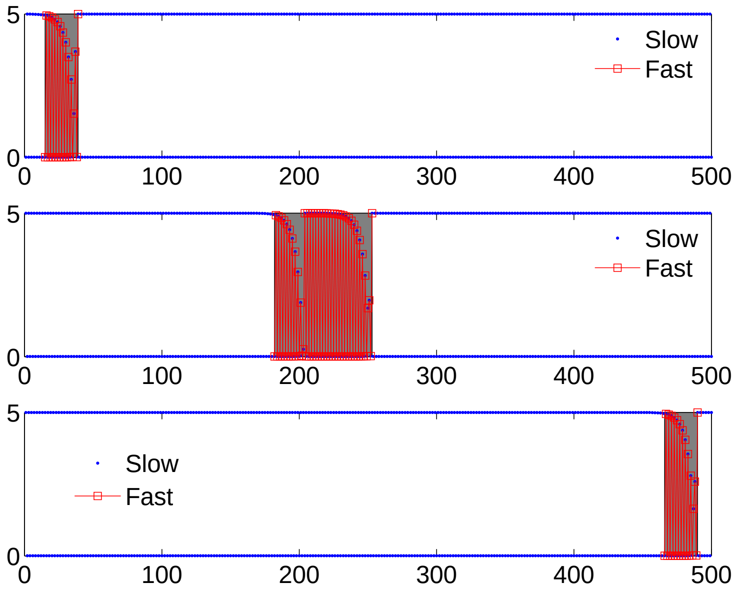


Figure 4: Solution of the inverter chain problem (13) at (top) $t = 15$ s, (middle) $t = 60$ s, (bottom) $t = 120$ s for $\Upsilon := 100$, respectively.

context can be found in [31; 29; 2; 20]. In this setting, the inverter-chain problem models the propagation of a signal that is injected through the first inverter after five seconds into the simulation and completely removed after 17 seconds (13b). This signal propagates through the chain, reaching the last inverter after about 120 seconds. The numerical error is estimated by using a reference run with relatively small time step. In every instance we estimate the maximum error at every time step, which amounts to taking the L_∞-norm in space-time.

We follow [2] and set the fast time scale to be adaptively selected based on the magnitude of the right-hand side of ODE (13), specifically $|RHS\{(13a)\}| < 0.05$. In this setting we obtain on average about 60 inverters in the fine time scale, which represents about 12% of total inverters. This region is represented in Figure 4 with a solid line and shadowed background at different times. We note that the fine region can be further reduced by coarsening the time grid in the middle of the traveling signal. In contrast with [2; 31], we consider only one contiguous refined region and do not address telescopic grids, which are certainly possible but not necessary to illustrate the convergence behavior of the methods under consideration.

We estimate the work involved in integrating the solution to the final time by the number of inverters in each scale (denoted here as n_{fast} and n_{slow}) and their corresponding computational cost per step:

$$\text{work} = \sum_{k=0}^{T/H} \frac{T}{H/m} n_{\text{fast}}(t_k) + \frac{T}{H} n_{\text{slow}}(t_k)\,,\ n = n_{\text{fast}} + n_{\text{slow}}\,. \tag{14}$$

This metric is used to compare the computational cost of extrapolation methods, and it should be interpreted termwise in the extrapolation tableau. The difference between single-rate and multirate schemes appears only in the base method. We therefore claim that this criterion is a fair estimate of the total work that avoids issues related to particular implementations or architectures.

The numerical errors for the stiff inverter-chain problem are shown in Table 6, when using the single rate and multirate extrapolated linearly implicit methods with different H and multirate scheme m, as well as terms in the extrapolation table. These results are illustrated in Figure 5, where we show the max-norm errors vs the measured work (14). We compare the extrapolated multirate (MREx) and single rate (SREx) linearly implicit schemes introduced in this study. We also show the estimated cost with a parallel implementation [26] for the single rate (Par SREx) and multirate (Par MREx). The parallel time is estimated by the amount of work needed for the most costly sequential calculation, that is, the last row needed to compute the solution. Significant computational work can be saved by using multirate schemes; moreover, parallelizing these extrapolated methods may yield an additional performance boost, as predicted by our results. We remark that using a larger time-step than considered in our experiments may lead to poor accuracy and instabilities in the extrapolation methods in general because of discontinuities in the original problem. This behavior is visible in Fig. 5 in the sudden jump in work as the error decreases. This aspect disadvantages efficient high-order methods that can take large steps; however, the robustness of the extrapolation method makes it perform well even for such a problem.

In Fig. 5 we show also the results reported by Savcenco et al. [31] by using a multirate Rosenbrock 2 (MR Ros-2) scheme. A direct comparison between the multirate approach introduced in this study and MR Ros-2 is difficult because time adaptivity and telescopic refinement of the temporal grid are used [31]. Our approach is based on a fixed time step throughout the simulation, and we use one level of refinement that is contiguous in space. Based on the numerical results obtained in this study, we conclude that both extrapolated multirate and MR Ros-2 methods are comparable in terms of performance, although the methods introduced in this study are arguably disatvantaged by the arguments given above. A more focused study on the implementation and computational challenges presented by both strategies can answer the question of which method works best for this problem.

In all our numerical experiments, we note that multirate schemes perform better than their single-rate counterparts. The optimal choice of rate and terms resolved in the extrapolation tableau is problem dependent and an open research problem that we do not address in this study.

Table 6: Tableau of max-norm errors illustrated in Figure 5 and corresponding to different extrapolation terms for the stiff ($\Gamma = 100$) inverter-chain problem solved with the single rate and multirate extrapolated linear implicit methods. The work measure is based on (14) (K=10^3).

Single rate - serial

Err	Work	$T_{j,k}$	H
1.14e+00	19500K	T_{31}	2.0e-02
9.59e-01	48750K	T_{51}	2.0e-02
8.73e-01	32500K	T_{42}	2.0e-02
7.63e-01	32500K	T_{44}	2.0e-02
5.70e-01	78000K	T_{32}	5.0e-03
1.54e-01	182000K	T_{72}	1.0e-02
1.34e-01	130000K	T_{42}	5.0e-03
5.57e-02	195000K	T_{52}	5.0e-03
3.88e-02	273000K	T_{62}	5.0e-03

Single rate - parallel (estimated)

Err	Work	$T_{j,k}$	H
1.14e+00	9750K	T_{31}	2.0e-02
8.73e-01	13000K	T_{42}	2.0e-02
7.63e-01	13000K	T_{44}	2.0e-02
5.70e-01	39000K	T_{32}	5.0e-03
1.54e-01	45500K	T_{72}	1.0e-02
1.34e-01	52000K	T_{42}	5.0e-03
5.57e-02	65000K	T_{52}	5.0e-03
3.88e-02	78000K	T_{62}	5.0e-03
3.08e-02	91000K	T_{72}	5.0e-03

Multirate - serial

Err	Work	$T_{j,k}$	H	m
1.87e+00	789K	T_{33}	5.0e-01	26
1.37e+00	1301K	T_{43}	5.0e-01	14
6.74e-01	1303K	T_{44}	5.0e-01	12
3.72e-01	23972K	T_{22}	1.0e-02	4
2.15e-01	28444K	T_{22}	1.0e-02	6
1.40e-01	32915K	T_{22}	1.0e-02	8
7.05e-02	41859K	T_{22}	1.0e-02	12
5.34e-02	46331K	T_{22}	1.0e-02	14
2.90e-02	47416K	T_{44}	2.0e-02	6
1.33e-02	54874K	T_{44}	2.0e-02	8
8.16e-03	62332K	T_{44}	2.0e-02	10
6.13e-03	65835K	T_{33}	1.0e-02	8
4.84e-03	69789K	T_{44}	2.0e-02	12
2.49e-03	74781K	T_{33}	1.0e-02	10
6.62e-04	83726K	T_{33}	1.0e-02	12
5.55e-04	114971K	T_{43}	1.0e-02	8
4.82e-04	160614K	T_{53}	2.0e-02	22

Multirate - parallel (estimated)

Err	Work	$T_{j,k}$	H	m
1.87e+00	395K	T_{33}	5.0e-01	26
1.37e+00	521K	T_{43}	5.0e-01	14
6.74e-01	521K	T_{44}	5.0e-01	12
5.03e-01	13970K	T_{55}	5.0e-02	12
3.62e-01	15175K	T_{44}	2.5e-02	6
5.04e-02	16763K	T_{65}	5.0e-02	12
2.90e-02	18967K	T_{44}	2.0e-02	6
1.49e-02	20349K	T_{66}	5.0e-02	16
1.33e-02	21950K	T_{44}	2.0e-02	8
8.16e-03	24933K	T_{44}	2.0e-02	10
4.81e-03	24938K	T_{54}	2.5e-02	10
2.27e-03	30898K	T_{44}	2.0e-02	14
1.41e-03	33881K	T_{44}	2.0e-02	16
9.71e-04	36864K	T_{44}	2.0e-02	18
6.62e-04	41863K	T_{33}	1.0e-02	12
5.55e-04	45989K	T_{43}	1.0e-02	8
4.82e-04	53538K	T_{53}	2.0e-02	22

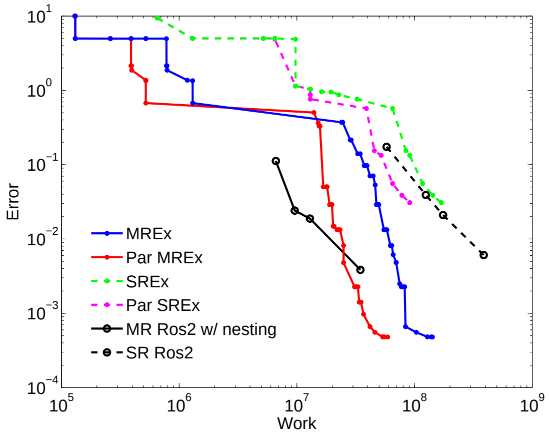


Figure 5: Max-norm errors for the inverter chain problem vs the measured work (14). We used extrapolated multirate (MREx) and single-rate (SREx) linearly implicit schemes introduced in this study and multirate Rosenbrock 2 schemes described in [31]. We also show the estimated cost with a parallel implementation for the single rate (Par SREx) and multirate (Par MREx) [26].

6 Concluding Remarks

Multirate methods are effective schemes for solving multiscale problems. In this manuscript we present extrapolated multirate implicit and explicit discretization methods that allow us to efficiently solve problems that have multiple scales. We propose two extrapolation methods that are based on multirate forward and linearly implicit Euler schemes. These methods have a small implementation cost and can easily reach high orders of accuracy.

The extrapolation method by itself represents a sequence of embedded methods, which can be used for step-size control and variable-order approaches because of their trivial extension to higher orders. Naïve implementations of extrapolation methods are typically less efficient than Runge-Kutta or linear multistep schemes. However, the extrapolation methods can be parallelized easily [26]. Each entry in the first extrapolation tableau column ($T_{i,1}$) can be computed independently. Moreover, the cost is linearly increasing, and thus each entry can be optimally scheduled on multiprocessor or multicore architectures. This approach could lead to more efficient overall implementations. All these features are inherited by our multirate extensions as well.

The extrapolated multirate forward Euler method shows a slight degradation of the linear stability region. In practice, we consider that the increased efficiency of the multirate method outweighs this minor drawback. We also note that this aspect has not been observed in our numerical experiments.

In this study we have explored several multirate strategies using a fixed time step and one level of nesting. Robust adaptivity in time, order, and rate as well as telescopic refinement is an open research question. Nevertheless, through the numerical investigation of the linear stability we determine that the extrapolated multirate linearly implicit method performs well for nonstiff problems and for stiff problems with relaxed coupling among components.

Acknowledgements

We thank the reviewers for their constructive critique and suggestions, which made this into a better paper. We also thank Valeriu Savcenco for his help with the results reported in [31].

References

[1] A. Aitken, *On interpolation by iteration of proportional parts without the use of differences*, Proc. Edinburgh Math. Soc., 3 (1932), pp. 56–76.

[2] A. Bartel and M. Günther, *A multirate W-method for electrical networks in state-space formulation*, Journal of Computational and Applied Mathematics, 147 (2002), pp. 411–425.

[3] K. Burrage, *Parallel and sequential methods for ordinary differential equations*, Oxford University Press, 1995.

[4] E. Constantinescu and A. Sandu, *Multirate timestepping methods for hyperbolic conservation laws*, Journal of Scientific Computing, 33 (2007), pp. 239–278.

[5] ——, *Extrapolated implicit-explicit time stepping*, SIAM Journal on Scientific Computing, 31 (2010), pp. 4452–4477.

[6] E. Constantinescu and A. Sandu, *On extrapolated multirate methods*, in Progress in Industrial Mathematics at ECMI 2008, H.-G. Bock, F. Hoog, A. Friedman, A. Gupta, H. Neunzert, W. R. Pulleyblank, T. Rusten, F. Santosa, A.-K. Tornberg, V. Capasso, R. Mattheij, H. Neunzert, O. Scherzer, A. D. Fitt, J. Norbury, H. Ockendon, and E. Wilson, eds., vol. 15 of Mathematics in Industry, Springer Berlin Heidelberg, 2010, pp. 341–347. 10.1007/978-3-642-12110-4_52.

[7] E. Constantinescu, A. Sandu, and G. Carmichael, *Modeling atmospheric chemistry and transport with dynamic adaptive resolution*, Computational Geosciences, 12 (2008), pp. 133–151.

[8] C. Dawson and R. Kirby, *High resolution schemes for conservation laws with locally varying time steps*, SIAM Journal on Scientific Computing, 22 (2001), pp. 2256–2281.

[9] P. Deuflhard, *Order and stepsize control in extrapolation methods*, Numerische Mathematik, 41 (1983), pp. 399–422.

[10] ——, *Recent progress in extrapolation methods for ordinary differential equations*, SIAM Review, 27 (1985), pp. 505–535.

[11] P. Deuflhard, E. Hairer, and J. Zugck, *One-step and extrapolation methods for differential-algebraic systems*, Numer. Math., 51 (1987), pp. 501–516.

[12] C. Engstler and C. Lubich, *Multirate extrapolation methods for differential equations with different time scales*, Computing, 58 (1997), pp. 173–185.

[13] M. Gasca and T. Sauer, *Polynomial interpolation in several variables*, Advances in Computational Mathematics, 12 (2000), pp. 377–410.

[14] C. Gear and D. Wells, *Multirate linear multistep methods*, BIT, 24 (1984), pp. 484–502.

[15] W. Gragg and H. Stetter, *Generalized multistep predictor-corrector methods*, J. ACM, 11 (1964), pp. 188–209.

[16] M. Günther, A. Kværnø, and P. Rentrop, *Multirate partitioned Runge-Kutta methods*, BIT, 41 (2001), pp. 504–514.

[17] M. Günther and P. Rentrop, *Multirate ROW-methods and latency of electric circuits*, Applied Numerical Mathematics, 13 (1993), pp. 83–102.

[18] E. Hairer, S. Nørsett, and G. Wanner, *Solving Ordinary Differential Equations I: Nonstiff Problems*, Springer, 1993.

[19] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*, Springer, 1993.

[20] W. Hundsdorfer and V. Savcenco, *Analysis of a multirate theta-method for stiff ODEs*, Applied Numerical Mathematics, 59 (2009), pp. 693–706.

[21] R. Kirby, *On the convergence of high resolution methods with multiple time scales for hyperbolic conservation laws*, Mathematics of Computation, 72 (2002), pp. 1239–1250.

[22] A. Kværnø, *Stability of multirate Runge-Kutta schemes*, International Journal of Differential Equations and Applications, 1 (2000), pp. 97–105.

[23] A. Kværnø and P. Rentrop, *Low order multirate Runge-Kutta methods in electric circuit simulation*, Preprint no. 99/1, IWRMM, University of Karlsruhe (1999).

[24] E. Neville, *Iterative interpolation*, J. Indian Math. Soc., 20 (1934), pp. 87–120.

[25] A. Prothero and A. Robinson, *On the stability and accuracy of one-step methods for solving stiff systems of ordinary differential equations*, Mathematics of Computation, 28 (1974), pp. 145–162.

[26] T. Rauber and G. Rünger, *Load balancing schemes for extrapolation methods*, Concurrency: Practice and Experience, 9 (1997), pp. 181–202.

[27] J. Rice, *Split Runge-Kutta methods for simultaneous equations*, Journal of Research of the National Institute of Standards and Technology, 64 (1960), pp. 151–170.

[28] A. Sandu and E. Constantinescu, *Multirate explicit Adams methods for time integration of conservation laws*, Journal of Scientific Computing, 38 (2009), pp. 229–249.

[29] V. Savcenco, *Multirate numerical integration for ordinary differential equations*, PhD thesis, Universiteit van Amsterdam, 2008.

[30] V. Savcenco, W. Hundsdorfer, and J. Verwer, *A multirate time stepping strategy for parabolic PDE*, Tech. Report MAS-E0516, Centrum voor Wiskundeen Informatica, 2005.

[31] ——, *A multirate time stepping strategy for stiff ODEs*, BIT, 47 (2007), pp. 137–155.

[32] M. Schlegel, O. Knoth, M. Arnold, and R. Wolke, *Multirate Runge-Kutta schemes for advection equations*, Journal of Computational and Applied Mathematics, 226 (2009), pp. 345–357.

[33] S. Skelboe, *Stability properties of backward differentiation multirate formulas*, Applied Numerical Mathematics, 5 (1989), pp. 151–160.

[34] H.-Z. Tang and G. Warnecke, *A class of high resolution schemes for hyperbolic conservation laws and convection-diffusion equations with varying time and space grids*, Journal on Computational Mathematics, 24 (2006), pp. 121–140.